1984

Interactive computer program for the analysis and design of linear time invariant systems.

Ismail, Habib.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

INTERACTIVE COMPUTER PROGRAM
FOR THE
ANALYSIS AND DESIGN OF LINEAR
TIME INVARIANT SYSTEMS

by

Habib Ismail

December 1984

Thesis Advisor: G. J. Thaler

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) Interactive Computer Program for the Analysis and Design of Linear Time Invariant Systems	5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; December 1984
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Habib Ismail	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943	12. REPORT DATE December 1984
	13. NUMBER OF PAGES 62
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Interactive Program, Bode Plot, Linear Control Systems

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In this thesis, an interactive computer program for the analysis and design of time invariant unity feedback control systems is presented, using cascade or feedback or both types of compensation.

By using this program, the user is freed from the tedious, time consuming and error prone method of hand calculations, letting the computer handle these tasks

efficiently and speedily. The user can then concentrate fully on the placement of poles and zeroes of the compensator(s) used.

Design of control systems by classical methods being essentially a repetitive, trial and error procedure, this program greatly cuts down the turn around time and leads to faster, more satisfactory results.

Approved for public release; distribution is unlimited.

Interactive Computer Program for the Analysis and Design of Linear Time Invariant Systems

by

Habib Ismail
Lieutenant Commander, Pakistan Navy
B.E., University of Karachi, Pakistan, 1974

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1984

ABSTRACT

In this thesis, an interactive computer program for the analysis and design of time invariant unity feedback control systems is presented, using cascade or feedback or both types of compensation.

By using this program, the user is freed from the tedious, time consuming and error prone method of hand calculations, letting the computer handle these tasks efficiently and speedily. The user can then concentrate fully on the placement of poles and zeroes of the compensator(s) used.

Design of control systems by classical methods being essentially a repetitive, trial and error procedure, this program greatly cuts down the turn around time and leads to faster, more satisfactory results.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

During the past two decades, the scientific and engineering communities have witnessed an ever increasing role of the digital computer in the fields of research, development and analysis of systems. The computer, today, is being used to solve engineering problems, whose solution, until very recently required long and tedious procedures. Still, it is probably true that this machine has potential not fully recognized yet, which is why the attention of so many computer scientists/engineers is focused on devising more efficient and innovative operating procedures.

Control system design is one area where classical theory has been extensively developed and used. It is fair to say that even today, most analysis and design problems of linear, time invariant control systems can still be approached using the methods developed by Bode, Nyquist and others.

A totally new approach to the design of control systems became available with the development of optimal control theory and the state variable analysis. These methods have been intensively developed in the last 10-15 years, but now their weaknesses have been exposed too. The "states" of the plant may not necessarily represent physically measurable quantities, and consequently it may not be possible to

implement the results at all. Luenberger's observers, designed to overcome this problem, can at best provide estimates of the state trajectory. Furthermore, the optimal control approach to design relies very heavily on mathematical manipulation, providing little insight to the actual working of the plant; the only input of the designer being the form of the cost function.

An intelligent use of the speed and information processing ability of the digital computer, coupled with the reliable features of classical theory appear to be the best solution to the problem at hand. The classical approach to design, being essentially a trial and error method, if the order of the system is fairly high, the number of repetitive calculations and the time required to perform these calculations becomes prohibitively large, the assistance of the computer in such problems becomes indispensable.

The work in this thesis was to develop an interactive, user oriented computer program that would prompt the user to input the transfer function and cascade/feedback compensators. The program would then display on the IBM 3277 - Tektronix 618 dial screen terminals the Bode Plot of both magnitude and phase. The program could be repeatedly used, with the user having the option to change/modify the compensators, each time viewing the effect of his modifications on the screen until he arrives at a satisfactory solution.

II. CONTROL ENGINEERING ANALYSIS

A. GENERAL

A continuous time control system may be represented in one of the following forms:

a. Transfer functions

b. State equations

c. System block diagrams or signal flow graphs

Algorithms exist in almost every undergraduate control engineering text to convert the system representation from one given form to another. Gianniotis (Ref. 1) describes a simple method of converting from transfer function to state variable form in Chapter II.

The transfer function representation in its most general form is:

$$\frac{A_m S^m + A_{m-1} S^{m-1} + A_{m-2} S^{m-2} + \ldots\ldots\ A_o S^o}{B_n S^n + B_{n-1} S^{n-1} + B_{n-2} S^{n-2} + \ldots\ldots\ B_o S^o}$$

Usually the mathematical description of the system is found in the transfer function form in the literature. Analysis and design of control systems by classical methods also requires the representation of the system in this form.

This thesis does not address the problem of converting from one form of representation to another. It is assumed that any conversions necessary have already been performed and that the system is represented by its open loop transfer function.

B. PROBLEM FORMULATION

Each system design has its own unique characteristics, but in general the system has to meet some kind of performance standards. These performance standards are generally provided as numerical specifications. The first step in the design of a control system is to analyze the system by itself in the usual feedback loop configuration. This is usually referred to as the uncompensated system.

Analysis of the uncompensated system almost always shows that the system cannot meet some or all of the given performance standards. Usually, additional components have to be inserted in the system for the purpose of altering the performance of the system. These components are called compensators. Compensation is a two step procedure, in which additional components (compensators) are inserted to change the structure of the system, and these components are then adjusted until the performance characteristics are satisfied.

The theory of cascade and feedback compensation is discussed in detail by Thaler (Ref. 2) in Chapters 5 and 6. Only a brief discussion of the types of compensators is presented here.

Compensators used in the design of feedback control systems are generally classified as shown in the block diagrams of figure 1.

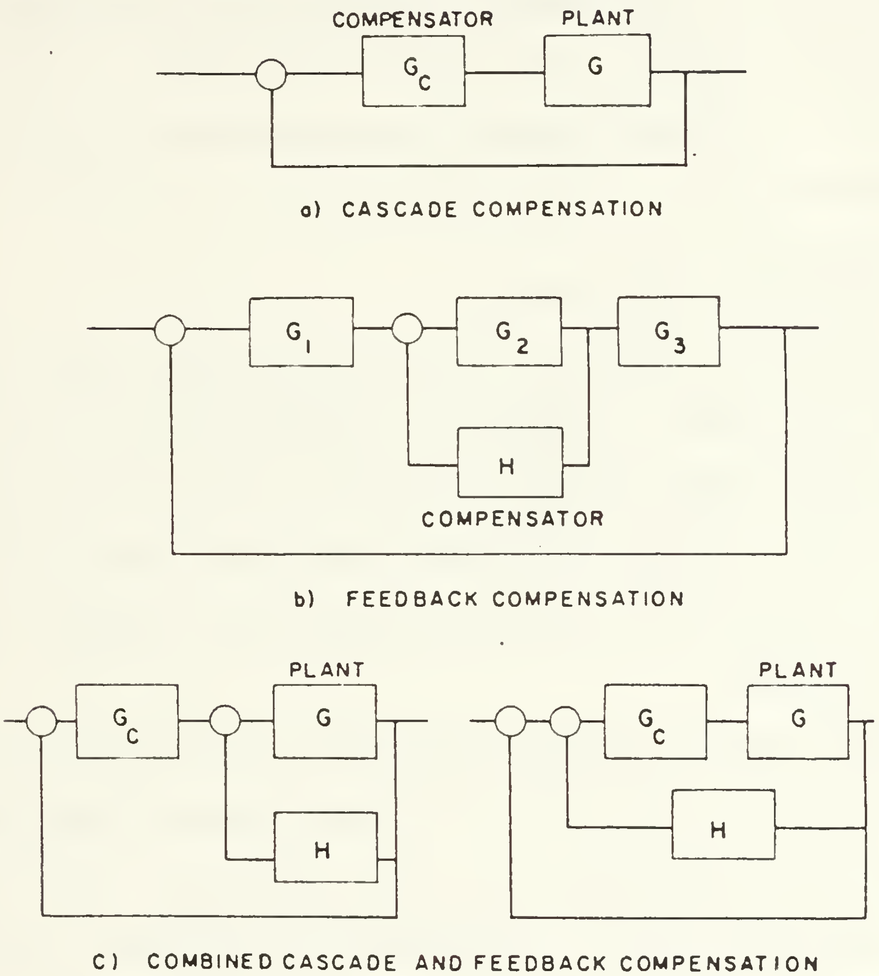

Figure 1. Classification of Compensation Structures

Cascade compensation may further be classified into two further types; a high pass filter usually called a phase lead compensator, and a low pass or phase lag compensator.

The selection of the type of compensator(s) to be used depends on a number of factors, the important ones being experience, personal preferences, availability, system constraints, etc. Unfortunately, there are no mathematical techniques to help in this selection process. Generally, one has to complete several designs and then choose the most appropriate one.

C. COMPUTER AIDED DESIGN

Of all the classical design methods available, the Bode diagram technique is generally considered to be the simplest, at the same time providing the most insight into system performance and behavior. The Bode design method may be used equally effectively both with cascade and feedback compensation schemes.

The computer program developed in this thesis displays on the terminal, initially, the Bode diagram (magnitude and phase) of the uncompensated open loop system. The user may then select the type of compensator to be used, and feed this information to the computer. The display changes, now showing the compensated system Bode Plot. This procedure may be repeated iteratively, with the computer updating the Bode Plot with each change made.

It was found on the average that turn around time for a typical third order system with two cascade compensators is less than five minutes.

III. PROGRAMMING CONSIDERATIONS

A. GENERAL

Any interactive software package, such as the one developed in this thesis must provide a simple yet unambigous means of data input and output. The data must be easy to interpret and apply to the problem at hand. Programs producing highly satisfactory results but requiring long studying time and/or special programming skills are of limited use only.

The intent of this thesis is to present such a program, Computer Aided Design of Linear Systems. Special care has been taken to develop the program so that the user has to invest very little time learning to use it.

B. MAIN FEATURES OF THE PROGRAM

The development of a user oriented interactive computer program in solving engineering problems requires a considerable amount of programming work, contributing significantly to its complexity and size.

The computer program, hereafter referred to as BODPLT consists of a main program, a number of programmer-composed subroutines, a few library functions/subroutines, and various subroutines of the DISSPLA graphics package. The entire program is written in the FORTRAN IV language.

In brief, the whole BODPLT package works as follows: A user, logged into the VM/CMS environment of the system from the dual screen terminals, issues the command DISSPLA BODPLT. The package then assumes control. The program begins its execution by interrogating the user and calling the appropriate subroutine accordingly. All programmer-composed subprograms are included in the main program titled BODPLT FORTRAN.

The BODPLT program has the following important features:

- runs of the VM/CMS time sharing system.

- interrogates the user in entering all problem specifications from the terminal.

- can handle up to a ninth order plant transfer function, six first order and one second order cascade filters.

- prompts the user to input the parameters of velocity feedback, acceleration feedback or approximate acceleration feedback as required.

- provides the solution in tabular form on the IBM 3277 screen and the Bode Plot on the TEK 618 terminal.

- can provide hard copy version of the problem specifications and tabular output by using the RECORD ON/ RECORD OFF execs, and of the BODE PLOT on the Tektronix printer where installed.

- allows problem specifications to be changed between runs.

C. PROGRAM DESCRIPTION

The main program is the coordination center which controls the calling of the supporting subroutines, in order to input the transfer function, cascade/feedback compensators and other necessary information. The tabular results and the Bode Plot are then displayed on the two screens respectively.

The main program as well as the accompanying subroutines can be found in Appendix A. They contain a sufficient number of comment cards to be self explanatory.

The following is a brief description of the performance and purpose of the various subroutines.

<u>NUMER</u> inputs the numerator of the plant transfer function.

<u>DENOM</u> inputs the denominator of the plant transfer function.

<u>CASCAD</u> inputs up to 6 first order lead/lag filters.

<u>SECAS</u> inputs the numerator of the second order band pass/band stop filter.

<u>SECASD</u> inputs the denominator of the second order filter.

<u>FETCH</u> determines the value of the radial frequency, w, at the origin of the x-axis.

<u>DECADE</u> determines the number of decades of frequency to be spanned.

<u>FEEDBK</u> inputs the various parameters of feedback compensators.

TITLES inputs the two lines of text as headings for the Bode Plot.

In addition, the main program handles the tasks of displaying the tabular output and the Bode Plot of the open and closed loop response of the system as required.

IV. PROGRAM PERFORMANCE INVESTIGATION

A. GENERAL

The program was tested by solving several linear control system design problems. Very satisfactory results were obtained in all cases with remarkable efficiency and speed. The only necessary condition is proper problem formulation. This is true for any interactive computer program. Once the program is used a couple of times, the user gets the necessary familiarity with its working.

The example problems presented below are used to demonstrate the performance and capabilities of the program. The examples can also help the user in formulating his own particular problem. The examples were selected from Thaler's 'Design of Feedback Systems' (Ref. 2)

B. EXAMPLE PROBLEMS

1. Example 4.1 : A Phase Lead Network

a. Problem Statement

A positioning system is single loop with unity feedback and forward transfer function

$$G(s) = \frac{5.0}{s(0.7s + 1)(0.3s + 1)}$$

The velocity constant is not to be decreased. Design a phase lead compensator which will provide M_{pw} less than 1.5.

b. Solution

The first step is to get the transfer function in the required form:

$$G(s) = \frac{5.0}{0.21S^3 + 1.0S^2 + 1.0S} \qquad (4.2)$$

The Bode Plot of the uncompensated system drawn using the program is shown on figure 2 and the tabular output on table 1. The uncompensated system has a phase margin of -20 degrees. To achieve a M_{pw} of less than 1.5, a phase margin of 44 degrees or more is required. Approximately 64 degrees of positive phase shift are therefore needed. Two sections of lead filter are introduced as given below:

$$G_c = \frac{(S/3.0 + 1)\ (S/1.5 + 1)}{(S/10.0 + 1)^2}$$

The compensated Bode Plot is given on figure 3 showing a phase margin of 35 degrees.

It may be pointed out that the final values for the lead filter poles and zeroes were arrived at after 3 iterations and this design problem was solved in less than 15 minutes.

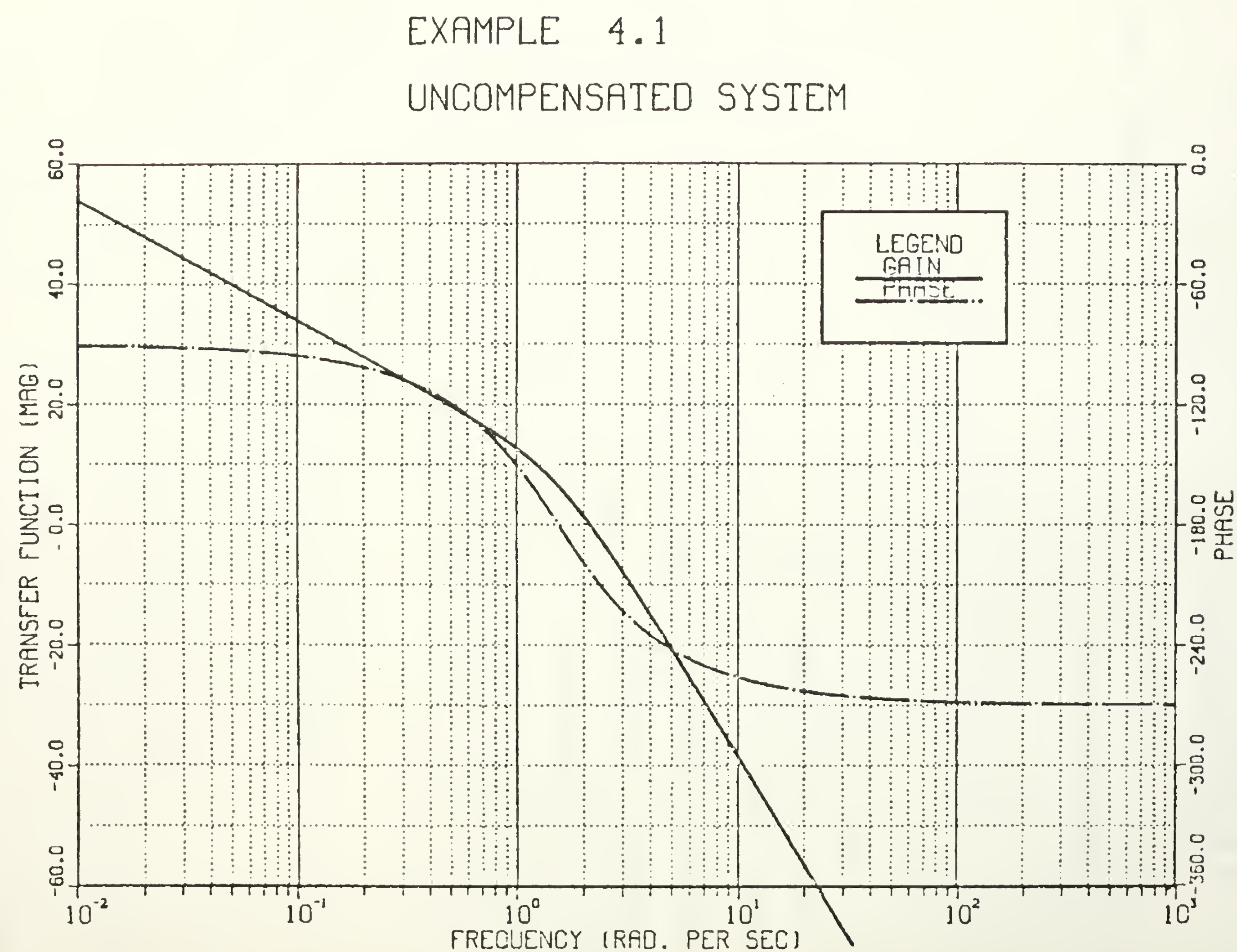

Figure 2. Uncompensated Bode Plot of Example 4.1

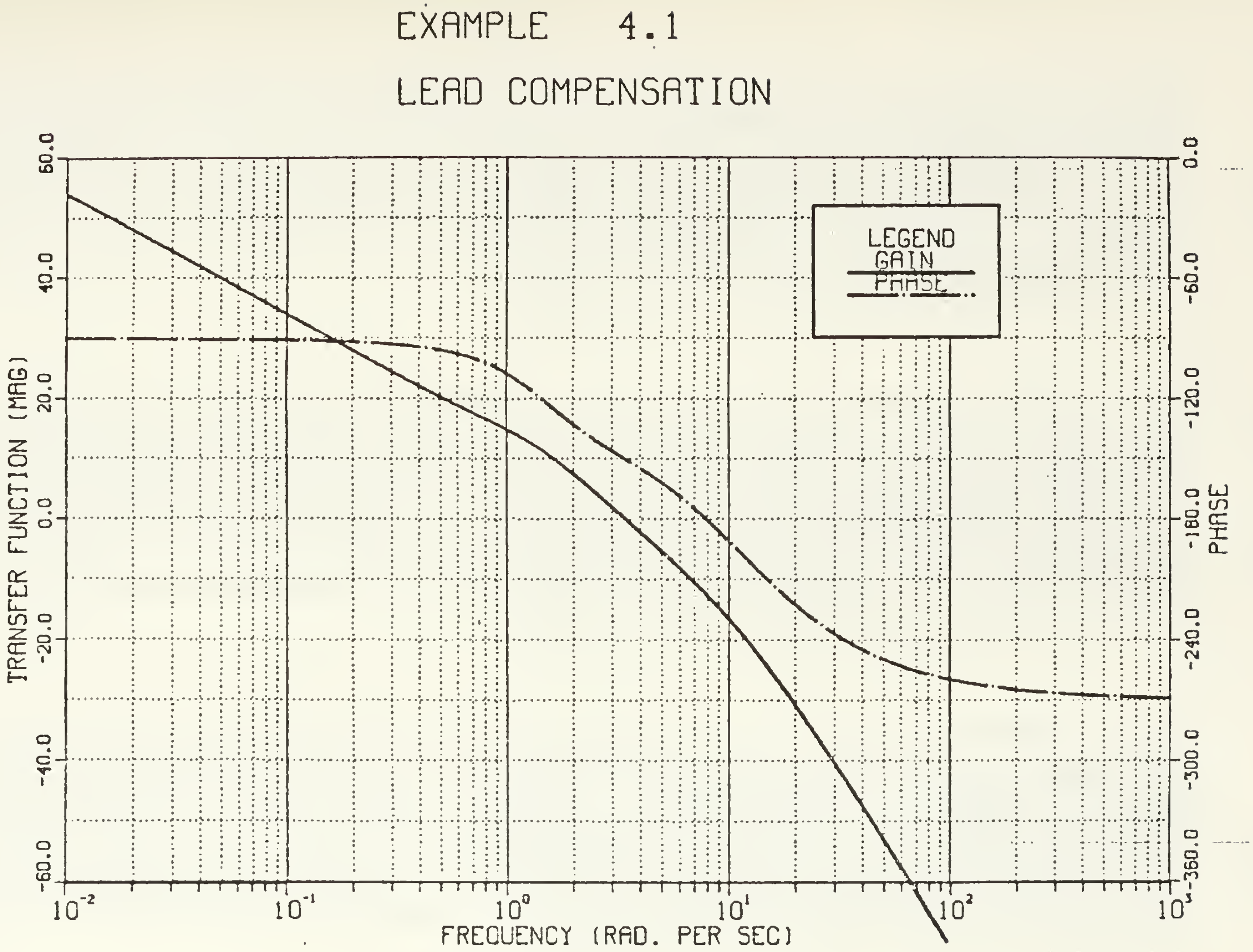

Figure 3. Lead Compensation Bode Plot of Example 4.1

Table 1. Tabular Output of Example 4.1

FREQ	MAGNITUDE	PHASE
0.100000E-01	0.539796E+02	-0.901134E+02
0.110776E-01	0.530907E+02	-0.901258E+02
0.122712E-01	0.522019E+02	-0.901395E+02
0.135936E-01	0.513130E+02	-0.901546E+02
0.150584E-01	0.504242E+02	-0.901714E+02
0.166810E-01	0.495354E+02	-0.901900E+02
0.184785E-01	0.486466E+02	-0.902106E+02
0.204697E-01	0.477579E+02	-0.902335E+02
0.226754E-01	0.468691E+02	-0.902588E+02
0.251188E-01	0.459804E+02	-0.902868E+02
0.278256E-01	0.450918E+02	-0.903179E+02
0.308240E-01	0.442032E+02	-0.903524E+02
0.341455E-01	0.433146E+02	-0.903905E+02
0.378249E-01	0.424262E+02	-0.904329E+02
0.419008E-01	0.415378E+02	-0.904798E+02
0.464158E-01	0.406496E+02	-0.905318E+02
0.514175E-01	0.397615E+02	-0.905896E+02
0.569580E-01	0.388736E+02	-0.906536E+02
0.630956E-01	0.379859E+02	-0.907247E+02
0.698946E-01	0.370985E+02	-0.908036E+02
0.774263E-01	0.362114E+02	-0.908913E+02
0.857695E-01	0.353247E+02	-0.909887E+02
0.950117E-01	0.344385E+02	-0.910972E+02
0.105250E+00	0.335529E+02	-0.912180E+02
0.116591E+00	0.326680E+02	-0.913526E+02
0.129155E+00	0.317841E+02	-0.915030E+02
0.143072E+00	0.309012E+02	-0.916711E+02
0.158489E+00	0.300196E+02	-0.918595E+02
0.175567E+00	0.291397E+02	-0.920711E+02
0.194486E+00	0.282617E+02	-0.923094E+02
0.215443E+00	0.273860E+02	-0.925787E+02
0.238658E+00	0.265130E+02	-0.928840E+02
0.264376E+00	0.256433E+02	-0.932318E+02
0.292864E+00	0.247775E+02	-0.936298E+02
0.324421E+00	0.239162E+02	-0.940876E+02
0.359381E+00	0.230600E+02	-0.946171E+02
0.398106E+00	0.222096E+02	-0.952334E+02
0.441004E+00	0.213655E+02	-0.959547E+02
0.488526E+00	0.205278E+02	-0.968035E+02
0.541168E+00	0.196966E+02	-0.978075E+02
0.599485E+00	0.188709E+02	-0.989995E+02
0.664081E+00	0.180487E+02	-0.100417E+03
0.735640E+00	0.172265E+02	-0.102103E+03
0.814911E+00	0.163984E+02	-0.104101E+03
0.902723E+00	0.155561E+02	-0.106450E+03
0.999999E+00	0.146882E+02	-0.109181E+03
0.110775E+01	0.137807E+02	-0.112300E+03
0.122712E+01	0.128181E+02	-0.115764E+03
0.135935E+01	0.117855E+02	-0.119567E+03
0.150583E+01	0.106723E+02	-0.123548E+03
0.166809E+01	0.947399E+01	-0.127600E+03
0.184785E+01	0.819452E+01	-0.131594E+03
0.204696E+01	0.684498E+01	-0.135427E+03
0.226754E+01	0.544112E+01	-0.139035E+03
0.251188E+01	0.399998E+01	-0.142404E+03
0.278255E+01	0.253684E+01	-0.145561E+03
0.308239E+01	0.106322E+01	-0.148567E+03
0.341454E+01	-0.413782E+00	-0.151495E+03
0.378247E+01	-0.189173E+01	-0.154429E+03
0.419007E+01	-0.337225E+01	-0.157444E+03

Table 1. (Contd.)

0.464157E+01	-0.486013E+01	-0.160608E+03
0.514174E+01	-0.636283E+01	-0.163974E+03
0.569579E+01	-0.788934E+01	-0.167576E+03
0.630955E+01	-0.944966E+01	-0.171432E+03
0.693945E+01	-0.110539E+02	-0.175540E+03
0.774261E+01	-0.127116E+02	-0.179883E+03
0.857692E+01	-0.144313E+02	-0.184433E+03
0.950116E+01	-0.162196E+02	-0.189133E+03
0.105250E+02	-0.180814E+02	-0.193936E+03
0.115591E+02	-0.200195E+02	-0.198782E+03
0.129155E+02	-0.220340E+02	-0.203612E+03
0.143072E+02	-0.241236E+02	-0.208369E+03
0.158489E+02	-0.262850E+02	-0.213003E+03
0.175567E+02	-0.285134E+02	-0.217469E+03
0.194485E+02	-0.308033E+02	-0.221735E+03
0.215443E+02	-0.331486E+02	-0.225776E+03
0.238658E+02	-0.355431E+02	-0.229574E+03
0.264375E+02	-0.379807E+02	-0.233121E+03
0.292863E+02	-0.404556E+02	-0.236416E+03
0.324421E+02	-0.429623E+02	-0.239462E+03
0.359380E+02	-0.454962E+02	-0.242267E+03
0.398106E+02	-0.480529E+02	-0.244840E+03
0.441004E+02	-0.506288E+02	-0.247195E+03
0.488526E+02	-0.532208E+02	-0.249344E+03
0.541167E+02	-0.558260E+02	-0.251302E+03
0.599483E+02	-0.584423E+02	-0.253083E+03
0.664080E+02	-0.610677E+02	-0.254700E+03
0.735639E+02	-0.637005E+02	-0.256168E+03
0.814910E+02	-0.663395E+02	-0.257498E+03
0.902722E+02	-0.689835E+02	-0.258703E+03
0.999998E+02	-0.716317E+02	-0.259793E+03

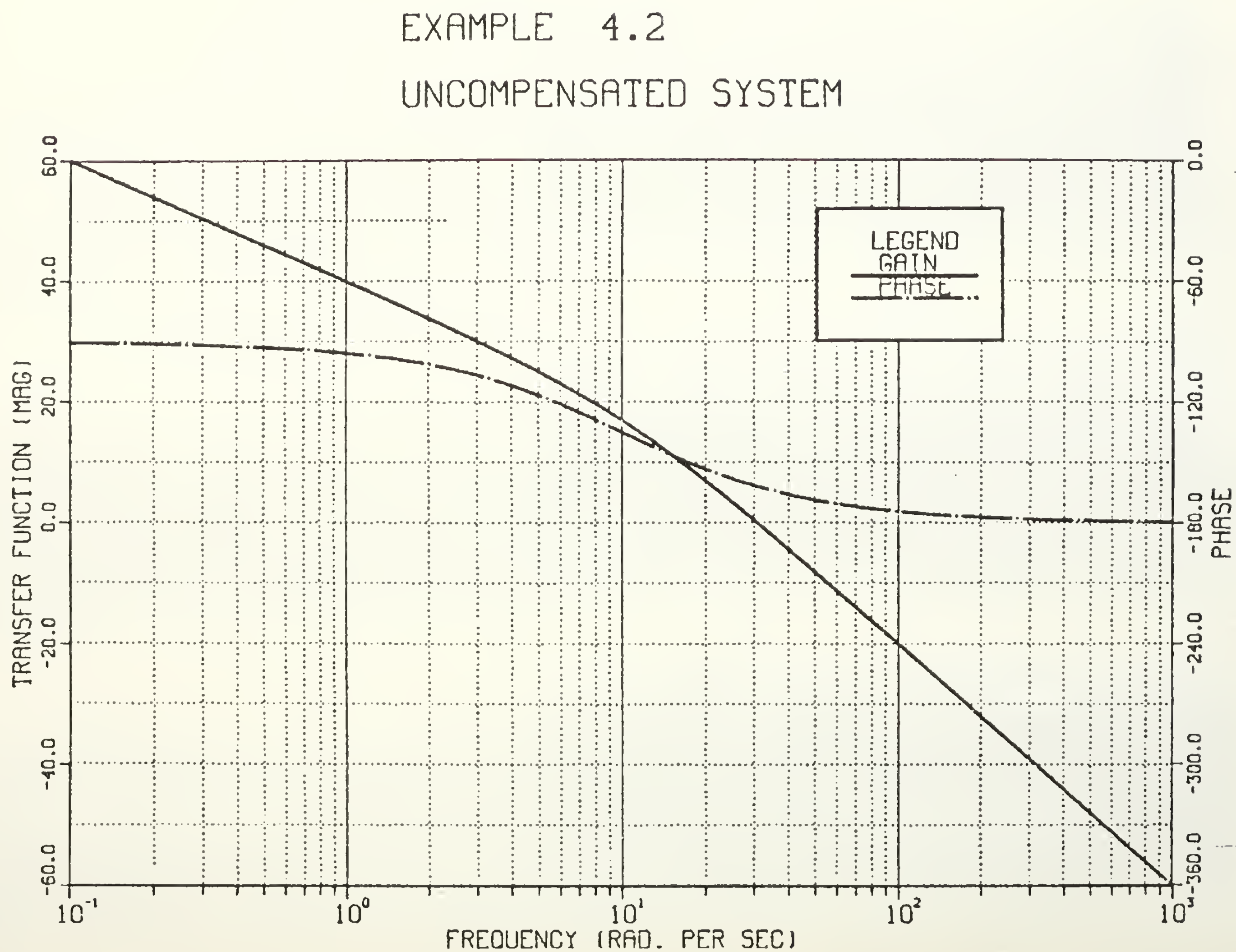

Figure 4. Uncompensated Bode Plot of Example 4.2

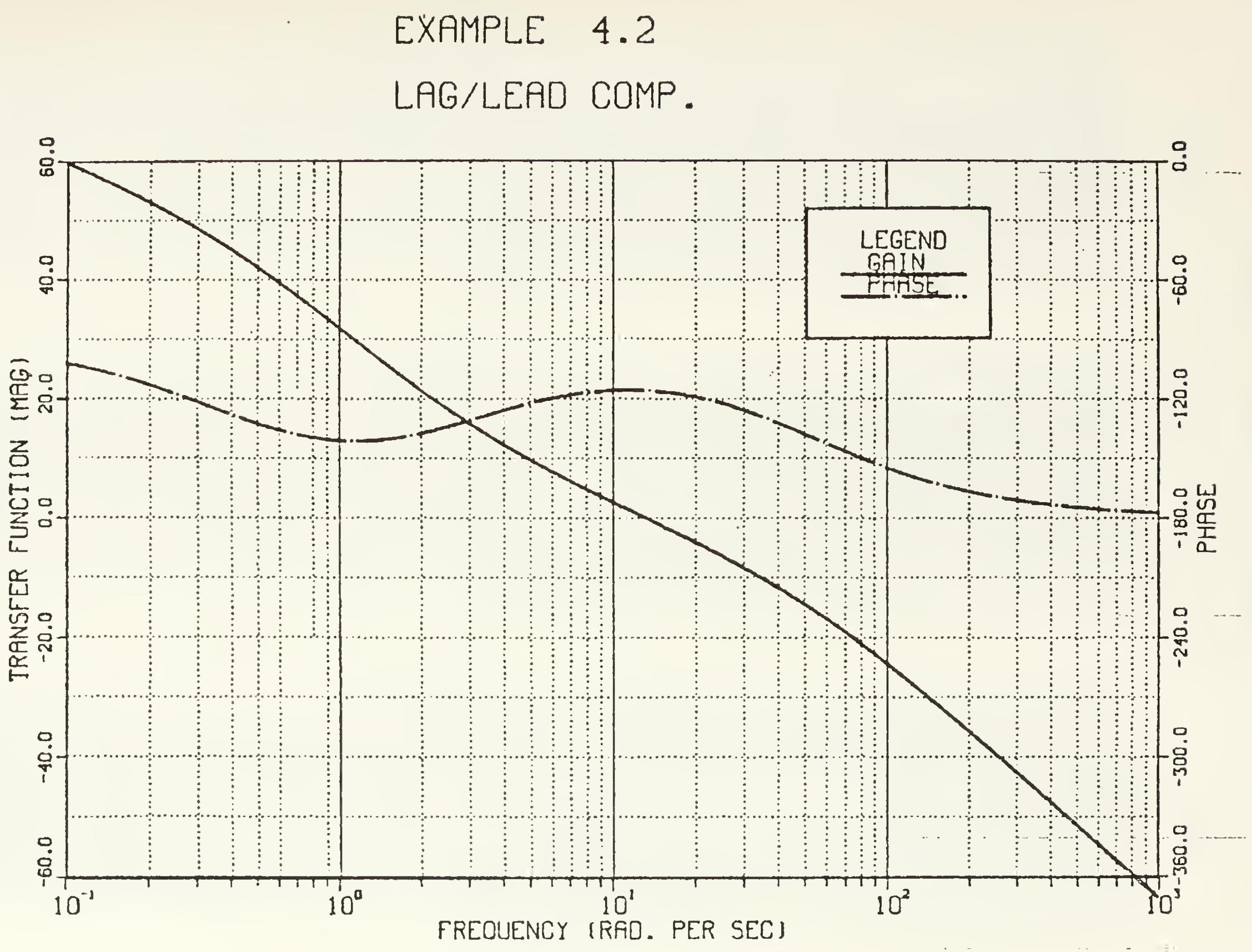

Figure 5. Lag/Lead Compensation in Example 4.2

Table 2. Tabular Output of Example 4.2

FREQ	MAGNITUDE	PHASE
0.100000E+00	0.598345E+02	-0.995141E+02
0.110776E+00	0.589089E+02	-0.100503E+03
0.122712E+00	0.579754E+02	-0.101586E+03
0.135935E+00	0.570325E+02	-0.102769E+03
0.150584E+00	0.560783E+02	-0.104058E+03
0.166810E+00	0.551106E+02	-0.105457E+03
0.184785E+00	0.541270E+02	-0.106968E+03
0.204697E+00	0.531249E+02	-0.108594E+03
0.226754E+00	0.521012E+02	-0.110331E+03
0.251188E+00	0.510529E+02	-0.112174E+03
0.278256E+00	0.499765E+02	-0.114115E+03
0.308240E+00	0.488687E+02	-0.116138E+03
0.341454E+00	0.477266E+02	-0.118226E+03
0.378248E+00	0.465474E+02	-0.120355E+03
0.419008E+00	0.453289E+02	-0.122491E+03
0.464158E+00	0.440698E+02	-0.124606E+03
0.514175E+00	0.427697E+02	-0.126663E+03
0.569580E+00	0.414292E+02	-0.128623E+03
0.630956E+00	0.400502E+02	-0.130449E+03
0.698946E+00	0.386356E+02	-0.132106E+03
0.774263E+00	0.371891E+02	-0.133560E+03
0.857695E+00	0.357155E+02	-0.134785E+03
0.950117E+00	0.342201E+02	-0.135758E+03
0.105250E+01	0.327090E+02	-0.136461E+03
0.116591E+01	0.311881E+02	-0.136884E+03
0.129155E+01	0.296640E+02	-0.137022E+03
0.143072E+01	0.281432E+02	-0.136877E+03
0.158489E+01	0.266320E+02	-0.136457E+03
0.175567E+01	0.251367E+02	-0.135775E+03
0.194485E+01	0.236631E+02	-0.134853E+03
0.215443E+01	0.222164E+02	-0.133716E+03
0.238658E+01	0.208014E+02	-0.132395E+03
0.264375E+01	0.194217E+02	-0.130927E+03
0.292864E+01	0.180801E+02	-0.129351E+03
0.324421E+01	0.167783E+02	-0.127709E+03
0.359381E+01	0.155166E+02	-0.126043E+03
0.398106E+01	0.142947E+02	-0.124393E+03
0.441004E+01	0.131107E+02	-0.122797E+03
0.488526E+01	0.119624E+02	-0.121289E+03
0.541168E+01	0.108466E+02	-0.119899E+03
0.599483E+01	0.975994E+01	-0.118652E+03
0.664081E+01	0.869860E+01	-0.117569E+03
0.735640E+01	0.765877E+01	-0.116666E+03
0.814911E+01	0.663658E+01	-0.115955E+03
0.902723E+01	0.562823E+01	-0.115445E+03
0.999999E+01	0.463006E+01	-0.115145E+03
0.110775E+02	0.363853E+01	-0.115060E+03
0.122712E+02	0.265011E+01	-0.115192E+03
0.135935E+02	0.166130E+01	-0.115547E+03
0.150583E+02	0.668658E+00	-0.116125E+03
0.166809E+02	-0.331396E+00	-0.116927E+03
0.184784E+02	-0.134252E+01	-0.117953E+03
0.204696E+02	-0.236847E+01	-0.119201E+03
0.226754E+02	-0.341315E+01	-0.120666E+03
0.251188E+02	-0.448058E+01	-0.122343E+03
0.278255E+02	-0.557482E+01	-0.124219E+03
0.308239E+02	-0.669992E+01	-0.126281E+03
0.341454E+02	-0.785960E+01	-0.128510E+03
0.378247E+02	-0.905738E+01	-0.130883E+03
0.419007E+02	-0.102961E+02	-0.133373E+03

Table 2. (Contd.)

0.464157E+02	-0.115780E+02	-0.135950E+03
0.514174E+02	-0.129043E+02	-0.138581E+03
0.569579E+02	-0.142753E+02	-0.141232E+03
0.630955E+02	-0.156905E+02	-0.143871E+03
0.698945E+02	-0.171484E+02	-0.146467E+03
0.774261E+02	-0.186466E+02	-0.148993E+03
0.857692E+02	-0.201822E+02	-0.151427E+03
0.950116E+02	-0.217522E+02	-0.153749E+03
0.105250E+03	-0.233528E+02	-0.155948E+03
0.116591E+03	-0.249806E+02	-0.158014E+03
0.129155E+03	-0.266322E+02	-0.159944E+03
0.143072E+03	-0.283043E+02	-0.161736E+03
0.158489E+03	-0.299941E+02	-0.163392E+03
0.175567E+03	-0.316988E+02	-0.164917E+03
0.194485E+03	-0.334161E+02	-0.166316E+03
0.215443E+03	-0.351439E+02	-0.167597E+03
0.238658E+03	-0.368805E+02	-0.168765E+03
0.264375E+03	-0.386245E+02	-0.169830E+03
0.292863E+03	-0.403745E+02	-0.170798E+03
0.324421E+03	-0.421296E+02	-0.171678E+03
0.359380E+03	-0.438888E+02	-0.172476E+03
0.398105E+03	-0.456513E+02	-0.173199E+03
0.441004E+03	-0.474166E+02	-0.173854E+03
0.488525E+03	-0.491842E+02	-0.174447E+03
0.541167E+03	-0.509536E+02	-0.174984E+03
0.599482E+03	-0.527247E+02	-0.175469E+03
0.664080E+03	-0.544969E+02	-0.175907E+03
0.735639E+03	-0.562702E+02	-0.176304E+03
0.814910E+03	-0.580443E+02	-0.176662E+03
0.902721E+03	-0.598190E+02	-0.176986E+03
0.999998E+03	-0.615944E+02	-0.177278E+03

3. <u>Example 4.3 : Velocity Feedback</u>

a. Problem Statement

A simple second-order servo is to be compensated with tachometer feedback. The forward transfer function is

$$G(S) = \frac{100.0}{S(S + 1)}$$

and the tachometer transfer function is $K_t S$. The tachometer is fed back around <u>all</u> of the forward gain. Using Bode diagram methods, set K_t to provide $M_{pw} = 1.3$.

b. Solution

Bode Plot for this system is shown on figure 6. The system has a phase margin of about 6 degrees. For $M_{pw} = 1.3$, a phase margin of 45 degrees is required. A rough graphical design on the uncompensated Bode Plot gives

$$1/H = 12.0/S \quad \text{or} \quad H = 0.08S$$

The Bode Plot for the compensated system is on figure 7, showing a phase margin of 50 degrees.

The close loop frequency response of this example, drawn using BODPLT is shown on figure 8.

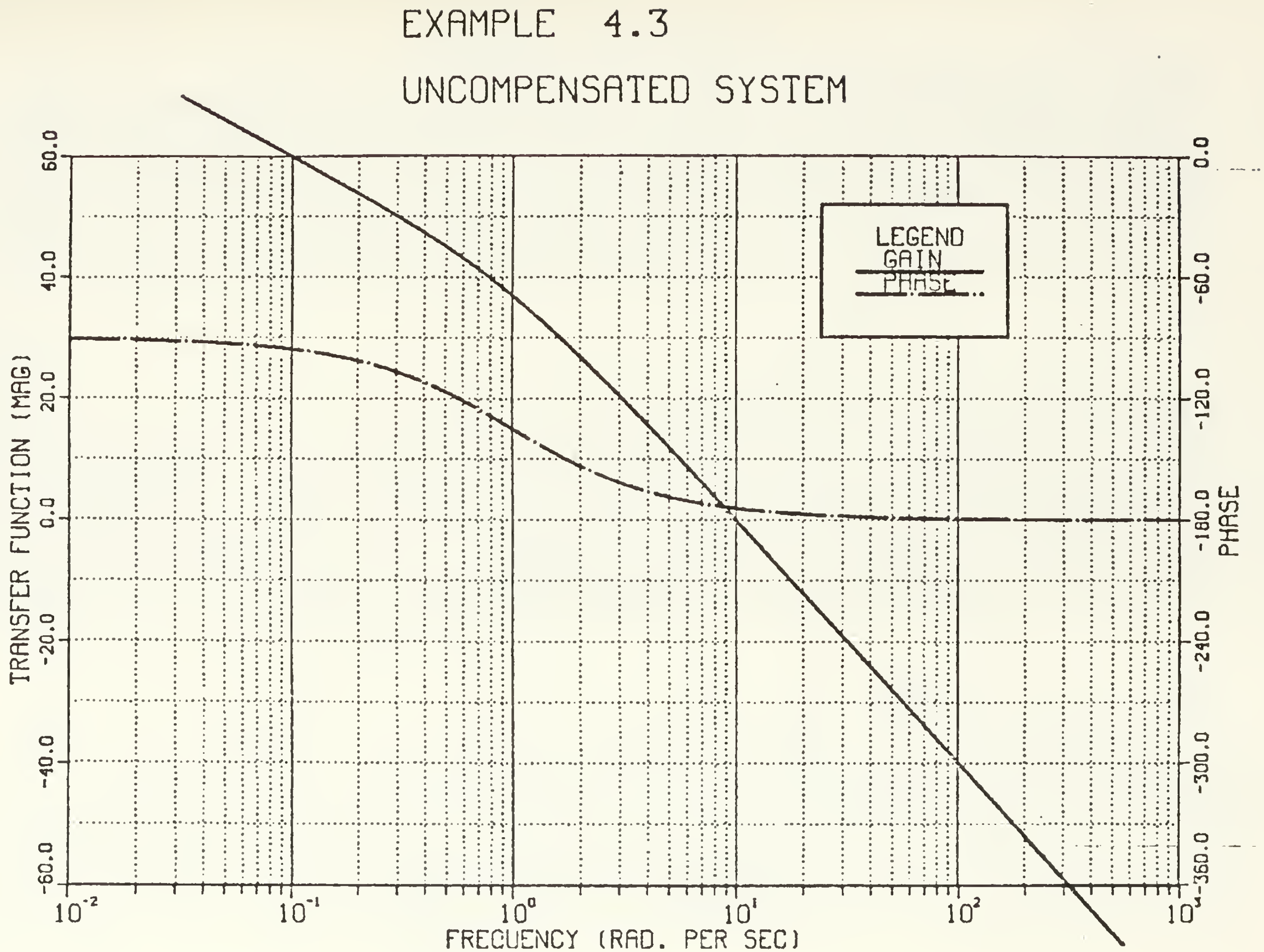

Figure 6. Uncompensated Bode Plot of Example 4.3

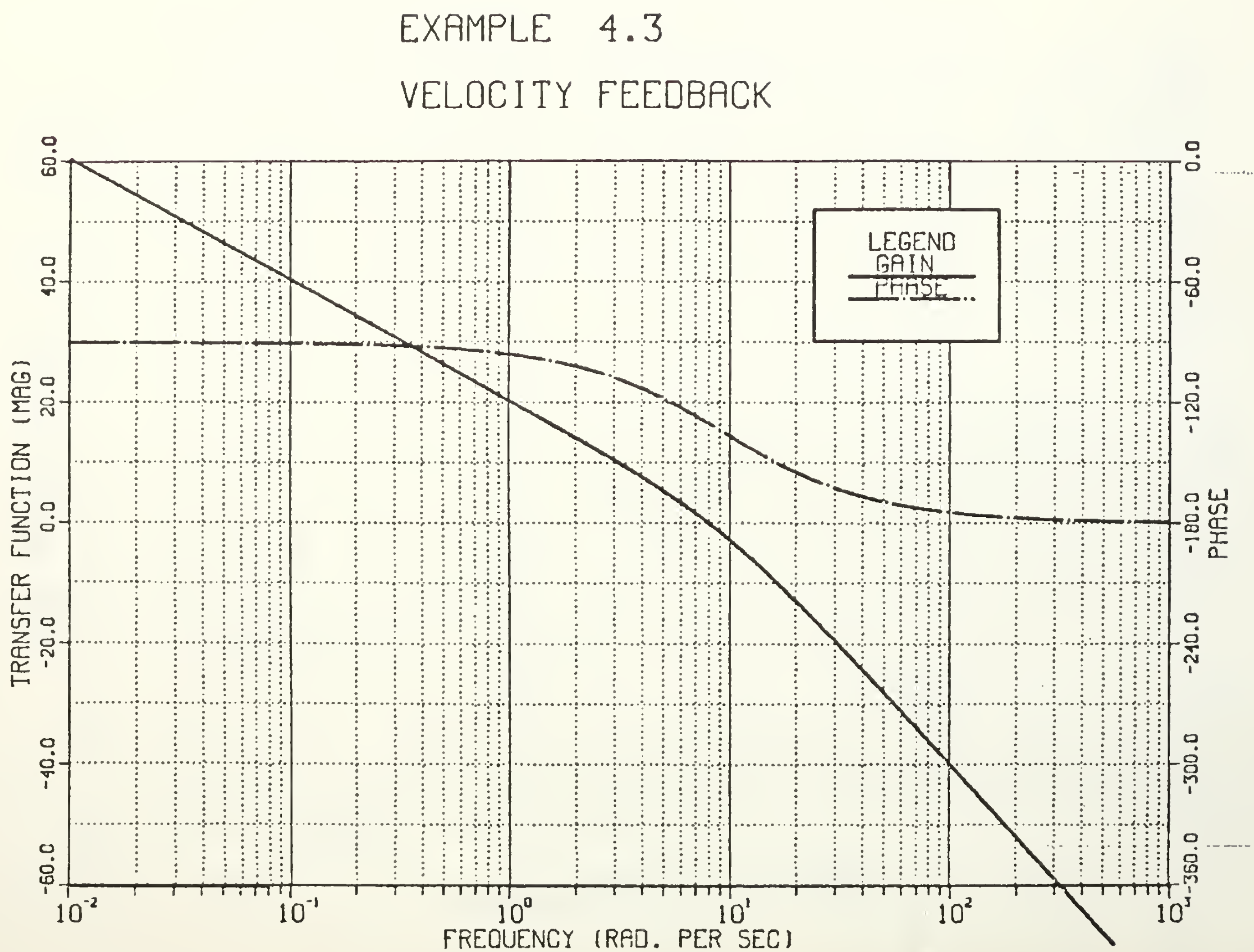

Figure 7. Velocity Feedback Used in Example 4.3

Table 3. Tabular Output of Example 4.3

FREQ	MAGNITUDE	PHASE
0.100000E-01	0.606303E+02	-0.900604E+02
0.113646E-01	0.595192E+02	-0.900688E+02
0.129155E-01	0.584081E+02	-0.900784E+02
0.146780E-01	0.572970E+02	-0.900892E+02
0.166810E-01	0.561859E+02	-0.901016E+02
0.189573E-01	0.550747E+02	-0.901156E+02
0.215443E-01	0.539636E+02	-0.901316E+02
0.244843E-01	0.528525E+02	-0.901497E+02
0.278256E-01	0.517414E+02	-0.901703E+02
0.316227E-01	0.506303E+02	-0.901936E+02
0.359381E-01	0.495192E+02	-0.902202E+02
0.408423E-01	0.484080E+02	-0.902504E+02
0.464159E-01	0.472969E+02	-0.902848E+02
0.527499E-01	0.461858E+02	-0.903238E+02
0.599483E-01	0.450746E+02	-0.903682E+02
0.681291E-01	0.439635E+02	-0.904185E+02
0.774263E-01	0.428523E+02	-0.904758E+02
0.879921E-01	0.417411E+02	-0.905409E+02
0.999998E-01	0.406298E+02	-0.906149E+02
0.113646E+00	0.395186E+02	-0.906989E+02
0.129155E+00	0.384073E+02	-0.907945E+02
0.146780E+00	0.372959E+02	-0.909030E+02
0.166810E+00	0.361845E+02	-0.910264E+02
0.189573E+00	0.350730E+02	-0.911666E+02
0.215443E+00	0.339613E+02	-0.913259E+02
0.244843E+00	0.328495E+02	-0.915069E+02
0.278255E+00	0.317376E+02	-0.917126E+02
0.316227E+00	0.306253E+02	-0.919463E+02
0.359381E+00	0.295127E+02	-0.922118E+02
0.408423E+00	0.283997E+02	-0.925134E+02
0.464157E+00	0.272862E+02	-0.928560E+02
0.527498E+00	0.261720E+02	-0.932451E+02
0.599483E+00	0.250568E+02	-0.936870E+02
0.681291E+00	0.239404E+02	-0.941886E+02
0.774262E+00	0.228226E+02	-0.947579E+02
0.879919E+00	0.217028E+02	-0.954037E+02
0.999996E+00	0.205804E+02	-0.961360E+02
0.113646E+01	0.194549E+02	-0.969657E+02
0.129155E+01	0.183252E+02	-0.979052E+02
0.146779E+01	0.171902E+02	-0.989675E+02
0.166809E+01	0.160484E+02	-0.100167E+03
0.189573E+01	0.148980E+02	-0.101520E+03
0.215443E+01	0.137367E+02	-0.103042E+03
0.244843E+01	0.125615E+02	-0.104748E+03
0.278255E+01	0.113691E+02	-0.106656E+03
0.316227E+01	0.101552E+02	-0.108778E+03
0.359381E+01	0.891480E+01	-0.111127E+03
0.408423E+01	0.764222E+01	-0.113708E+03
0.464157E+01	0.633108E+01	-0.116522E+03
0.527498E+01	0.497453E+01	-0.119560E+03
0.599482E+01	0.356567E+01	-0.122804E+03
0.681290E+01	0.209793E+01	-0.126224E+03
0.774261E+01	0.565706E+00	-0.129777E+03
0.879919E+01	-0.103506E+01	-0.133413E+03
0.999996E+01	-0.270649E+01	-0.137075E+03
0.113646E+02	-0.444846E+01	-0.140704E+03
0.129155E+02	-0.625853E+01	-0.144242E+03
0.146779E+02	-0.813240E+01	-0.147639E+03
0.166809E+02	-0.100643E+02	-0.150857E+03
0.189573E+02	-0.120476E+02	-0.153866E+03

0.215443E+02	-0.140753E+02	-0.156650E+03
0.244843E+02	-0.161408E+02	-0.159199E+03
0.278255E+02	-0.182376E+02	-0.161517E+03
0.316226E+02	-0.203602E+02	-0.163609E+03
0.359380E+02	-0.225037E+02	-0.165489E+03
0.408423E+02	-0.246639E+02	-0.167170E+03
0.464157E+02	-0.268375E+02	-0.168668E+03
0.527497E+02	-0.290217E+02	-0.169999E+03
0.599482E+02	-0.312143E+02	-0.171179E+03
0.681290E+02	-0.334134E+02	-0.172224E+03
0.774261E+02	-0.356177E+02	-0.173148E+03
0.879919E+02	-0.378259E+02	-0.173964E+03
0.999995E+02	-0.400373E+02	-0.174684E+03
0.113646E+03	-0.422511E+02	-0.175319E+03
0.129154E+03	-0.444668E+02	-0.175879E+03
0.146779E+03	-0.466840E+02	-0.176372E+03
0.166809E+03	-0.489023E+02	-0.176806E+03
0.189573E+03	-0.511215E+02	-0.177189E+03
0.215443E+03	-0.533413E+02	-0.177526E+03
0.244843E+03	-0.555617E+02	-0.177822E+03
0.278255E+03	-0.577825E+02	-0.178083E+03
0.316226E+03	-0.600037E+02	-0.178313E+03
0.359379E+03	-0.622250E+02	-0.178515E+03
0.408422E+03	-0.644466E+02	-0.178693E+03
0.464157E+03	-0.666683E+02	-0.178850E+03
0.527497E+03	-0.688902E+02	-0.178988E+03
0.599481E+03	-0.711121E+02	-0.179109E+03
0.681290E+03	-0.733341E+02	-0.179215E+03
0.774260E+03	-0.755561E+02	-0.179309E+03
0.879918E+03	-0.777782E+02	-0.179392E+03
0.999997E+03	-0.800003E+02	-0.179465E+03

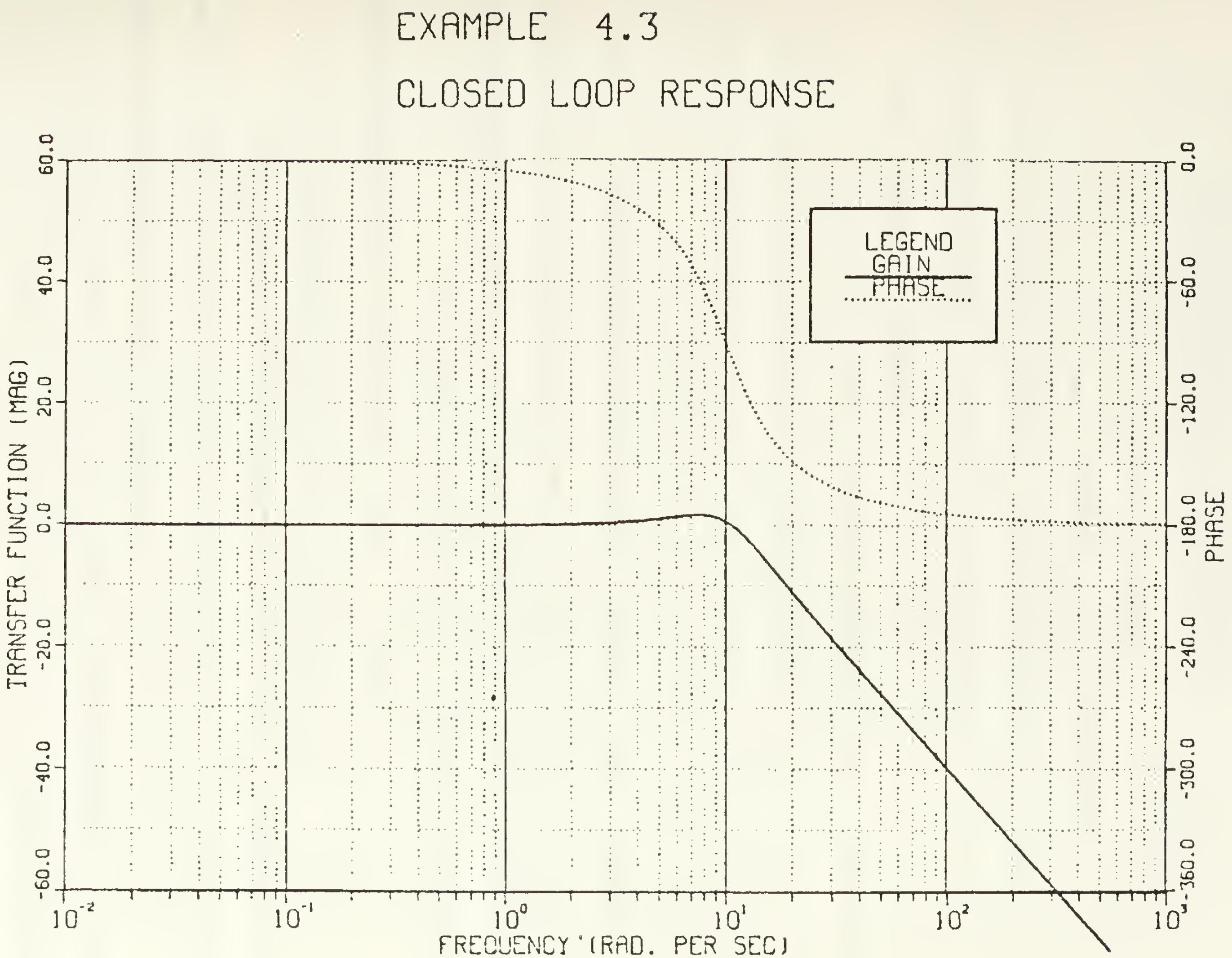

Figure 8. Closed Loop Response of Example 4.3

V. CONCLUSION AND RECOMMENDATIONS

A. CONCLUSION

The objective of this thesis was to develop an interactive user oriented computer program which would aid in solving control engineering problems using the Bode method of design. The presented program proves that frequency domain design of control systems using the digital computer as an aid is not only feasible but highly desirable. The results obtained are readily interpretable and provide good and meaningful insight into the problem.

The results obtained during the investigation of the program performance show that a complicated but well formulated problem can be solved with ease, and the solution is obtained with speed, accuracy and precision.

The entire program is less than 1000 lines, with a total of 9 subroutines. Every effort has been made to keep the program simple yet unambigous, so that the user has to invest very little time learning how to use it. Effort has also been made to minimize the use of the computer CPU time. However, expenditure of CPU time is to a large measure dependent upon:

(1) the order of the system.

(2) the number of iterations used in reaching a solution.

(3) the type and order of the compensator(s) used.

B. RECOMMENDATIONS

The program as presented in this thesis seems to be able to adequately satisfy the usual needs in a control system design problem. A number of useful extensions to the work developed in this thesis can be carried out. These are briefly discussed.

1. Curve Fitting

Although not specifically worked on in this thesis, the program can be used quite effectively for curve fitting purposes. This was demonstrated in the initial stages of the development of this program. The procedure is, by its very nature, iterative and therefore time consuming and cumbersome. However, the entire algorithm can be automated using a minimization subroutine, with the program outputting a polynomial to fit a given curve over a specified range of the independent variable.

2. Computer Selection of Compensators

A suitable minimization routine such as Box PLX can be incorporated into the program which could select the best possible location of poles and zeroes to meet given performance specifications. This would automate the entire Bode design procedure, the user then having the option of only specifying the type of compensation, i.e., cascade or feedback. It may however be pointed out that minimization

routines by the very nature of their operation are very time consuming and wasteful of CPU resources.

3. Root Locus

Most of the subroutines developed in this thesis are very general in nature and can be adapted quite easily to develop a similar interactive program for Root Locus plots.

4. Integrated Control System Design

No meaningful design of control systems is complete without finally analyzing its time domain performance. It is therefore considered highly desirable to incorporate into this program, an interactive Root Locus design procedure and then a time domain analysis of the compensated system. The entire package would then be an excellent teaching aid for control system design.

APPENDIX
PROGRAM LISTING

```
      SUBROUTINE NUMER (NA)
C
C     ****************************************
C     *  SUBROUTINE TO INPUT THE NUMERATOR   *
C     *  OF THE PLANT TRANSFER FUNCTION      *
C     ****************************************
C
C     *** VARIABLE DECLARATIONS ***
      INTEGER ANSWER,YES
      REAL NA
      DIMENSION NA(10)
      DATA YES/'Y'/
C
      DO 301 I=1,10
      NA(I) = 0.0
  301 CONTINUE
      CALL FRTCMS('CLRSCRN')
      WRITE(6,351)
      READ(5,376)N
      WRITE(6,352)N
      READ(5,377)ANSWER
      IF(ANSWER .EQ.  YES)GO TO 302
      GO TO 301
  302 CONTINUE
      N=N+1
  303 CONTINUE
C
      CALL FRTCMS('CLRSCRN')
      WRITE(6,360)
      DO 306  I=1,N
      J=I-1
  304 WRITE (6,353) J
      READ (5,378,END = 305) NA(I)
      GO TO 306
  305 REWIND 5
      CALL FRTCMS ('CLRSCRN')
      WRITE(5,356)
      GO TO 304
  306 CONTINUE
      CALL FRTCMS('CLRSCRN')
      DO 307 I  = 1,N
      J = N-I
      K = J+1
      WRITE(6,354) J,NA(K)
  307 CONTINUE
      WRITE(6,355)
      READ(5,379)ANSWER
      IF (ANSWER.EQ.YES) GO TO 308
```

```
      GO TO 3C3
308   CONTINUE
      CALL FRTCMS('CLRSCRN')
      RETURN
C
C**************  WRITE  FORMAT  STATEMENTS  *********************
C
351   FORMAT(///,7X,'ENTER CRDER OF  NUMERATOR  POLYNOMIAL AS A 1',
     *       ' DIGIT INTEGER')
352   FORMAT(//,7X,'ORDER OF NUMERATOR POLYNOMIAL IS ',I1,4X,
     *       'CORRECT ? (Y/N)')
353   FORMAT(//,7X,' ENTER COEFFICIENT OF S**(',I2,')')
354   FORMAT(/,' COEFFICIENT OF  S**',I2,' :       ',E14.6)
355   FORMAT(//,7X,' ARE THE ABOVE COEFFICIENTS CORRECT? (Y/N)')
356   FORMAT(//,7X,' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY',
     *       ' NUMBER.',/,7X,' TRY AGAIN.')
360   FORMAT(/,7X,'ALL NUMERATOR COEFFICIENTS MUST BE ENTERED ',
     *       'IN DECIMAL FORMAT')
C
C**************  READ  FORMAT  STATEMENTS  ************************
C
376   FORMAT(I1)
377   FORMAT(A1)
378   FORMAT(F14.4)
379   FORMAT(A1)
C
      END
```

```
      SUBROUTINE DENOM (DA)
C
C     *****************************************
C     *  SUBROUTINE TO INPUT THE DENOMINATOR  *
C     *  OF THE PLANT TRANSFER FUNCTION       *
C     *****************************************
C
C     *** VARIABLE DECLARATIONS ***
      INTEGER ANSWER,YES
      REAL DA
      DIMENSION DA(10)
      DATA YES/'Y'/
C
      DO 401 I=1,10
      DA(I) = 0.0
 401  CONTINUE
      CALL FRTCMS('CLRSCRN')
      WRITE(6,451)
      READ(5,476)N
      WRITE(6,452)N
      READ(5,477)ANSWER
      IF(ANSWER .EQ.  YES)GO TO 402
      GO TO 401
402   CONTINUE
      N=N+1
403   CONTINUE
C
      CALL FRTCMS('CLRSCRN')
      WRITE(6,460)
      DO 406 I=1,N
      J=I-1
404   WRITE (6,453) J
      READ (5,478,END = 405) DA(I)
      GO TO 406
405   REWIND 5
      CALL FRTCMS ('CLRSCRN')
      WRITE(5,456)
      GO TO 404
406   CONTINUE
      CALL FRTCMS('CLRSCRN')
      DO 407 I  = 1,N
      J = N-I
      K = J+1
      WRITE(6,454) J,DA(K)
407   CONTINUE
      WRITE(6,455)
      READ(5,479)ANSWER
      IF (ANSWER.EQ.YES) GO TO 408
```

```
      GO TO 403                                                         DENO0490
408   CONTINUE                                                          DENO0500
      CALL FRTCMS('CLRSCRN')                                            DENO0510
      RETURN                                                            DENO0520
C                                                                       DENO0530
C**************  WRITE  FORMAT  STATEMENTS  *********************        DENO0540
C                                                                       DENO0550
451   FORMAT(///,7X,'ENTER ORDER OF DENOMINATOR POLYNOMIAL AS A 1',     DENO0560
     *       ' DIGIT NUMBER')                                           DENO0570
452   FORMAT(//,7X,'ORDER OF DENOMINATOR POLYNOMIAL IS ',I1,4X,         DENO0580
     *       'CORRECT ? (Y/N)')                                         DENO0590
453   FORMAT(//,7X,' ENTER COEFFICIENT OF S**(',I2,')')                 DENO0600
454   FORMAT(/,' COEFFICIENT OF  S**',I2,' :        ',E14.6)            DENO0610
455   FORMAT(//,7X,' ARE THE ABOVE COEFFICIENTS CORRECT? (Y/N)')        DENO0620
456   FORMAT(//,7X,' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY',    DENO0630
     *       ' NUMBER.',/,7X,' TRY AGAIN.')                             DENO0640
460   FORMAT(//,7X,' ALL DENOMINATOR COEFFICIENTS MUST BE ENTERED ',    DENO0650
     *       'IN DECIMAL FORMAT')                                       DENO0660
C                                                                       DENO0670
C**************  READ  FORMAT  STATEMENTS  **************************** DENO0680
C                                                                       DENO0690
476   FORMAT(I1)                                                        DENO0700
477   FORMAT(A1)                                                        DENO0710
478   FORMAT(F14.4)                                                     DENO0720
479   FORMAT(A1)                                                        DENO0730
C                                                                       DENO0740
      END                                                               DENO0750
```

```
      SUBROUTINE CASCAD(CZS,CPS)
C
C     ******************************************
C     *   SUBROUTINE TO INPUT THE ZEROS AND    *
C     *   POLES OF THE CASCADE COMPENSATORS    *
C     ******************************************
C
C     *** VARIABLE DECLARATIONS ***
      INTEGER ANSWER,YES
      REAL CZS,CPS
      DIMENSION CZS(6),CPS(6)
      DATA YES/'Y'/
      DO 520 I = 1,6
      CZS(I)=1.0
      CPS(I)=1.0
520   CONTINUE
501   CONTINUE
      CALL FRTCMS('CLRSCRN')
      WRITE(6,551)
      READ(5,576)N
      WRITE(6,552)N
      READ(5,577)ANSWER
      IF(ANSWER .EQ.  YES)GO TO 503
      GO TO 501
503   CONTINUE
C
      CALL FRTCMS('CLRSCRN')
      WRITE(6,560)
      DO 506 I=1,N
504   WRITE (6,553) I
      READ (5,578,END = 505) CZS(I)
      GO TO 506
505   REWIND 5
      CALL FRTCMS ('CLRSCRN')
      WRITE(5,556)
      GO TO 504
506   CONTINUE
      CALL FRTCMS('CLRSCRN')
      DO 507 I  = 1,N
      WRITE(6,554) I,CZS(I)
507   CONTINUE
      WRITE(6,555)
      READ(5,577)ANSWER
      IF (ANSWER.EQ.YES) GO TO 508
      GO TO 503
508   CONTINUE
      DO 516 I=1,N
514   WRITE (6,563) I
```

```
      READ (5,578,END = 515) CPS(I)
      GO TO 516
515   REWIND 5
      CALL FRTCMS ('CLRSCRN')
      WRITE(5,556)
      GO TO 514
516   CONTINUE
      CALL FRTCMS('CLRSCRN')
      DO 517 I  = 1,N
      WRITE(6,564) I,CPS(I)
517   CONTINUE
      WRITE(6,555)
      READ(5,577)ANSWER
      IF (ANSWER.EQ.YES) GO TO 518
      GO TO 508
518   CONTINUE
      CALL FRTCMS('CLRSCRN')
      RETURN
C
C**************  WRITE  FORMAT  STATEMENTS  **********************
C
551   FORMAT(///,7X,'ENTER NUMBER OF CASCADE FILTERS AS A 1',
     *        ' DIGIT INTEGER')
552   FORMAT(//,7X,I1,' CASCADE FILTERS ARE BEING USED',4X,
     *        'CORRECT ? (Y/N)')
553   FORMAT(//,7X,' ENTER ZERO OF FILTER NUMBER ',I2,)
554   FORMAT(/,' ZERO NUMBER ',I2,' :          ',E14.6)
555   FORMAT(//,7X,' ARE THE ABOVE VALUES CORRECT? (Y/N)')
556   FORMAT(//,7X,' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY',
     *        ' NUMBER.',/,7X,' TRY AGAIN.')
560   FORMAT(//,7X,' VALUES OF POLES AND ZEROS OF THE CASCADE ',
     *        'FILTERS MUST BE ENTERED ',/,7X,' IN DECIMAL FORMAT')
563   FORMAT(//,7X,' ENTER POLE OF FILTER NUMBER ',I2,)
564   FORMAT(/,' POLE NUMBER ',I2,' :          ',E14.6)
C
C
C**************  READ  FORMAT  STATEMENTS  ************************
C
576   FORMAT(I1)
577   FORMAT(A1)
578   FORMAT(F12.5)
C
      END
```

```
      SUBROUTINE SECAS(CN)
C
C     ****************************************
C     *  SUBROUTINE TO INPUT THE NUMERATOR     *
C     *  OF   THE SECOND  ORDER COMPENSATOR    *
C     ****************************************
C
C     *** VARIABLE DECLARATIONS ***
      INTEGER ANSWER,YES
      REAL CN
      DIMENSION CN(3)
      DATA YES/'Y'/
C
C
      CALL FRTCMS('CLRSCRN')
301   CONTINUE
C
      WRITE(6,351)
      DO 306 I=1,3
      J=I-1
304   CONTINUE
      WRITE (6,353) J
      READ (5,378,END = 305) CN(I)
      GO TO 306
305   REWIND 5
      CALL FRTCMS ('CLRSCRN')
      WRITE(5,356)
      GO TO 304
306   CONTINUE
      CALL FRTCMS('CLRSCRN')
      DO 307 I  = 1,3
      J = 3-I
      K = J+1
      WRITE(6,354) J,CN(K)
307   CONTINUE
      WRITE(6,355)
      READ(5,379)ANSWER
      IF (ANSWER.EQ.YES) GO TO 308
      GO TO 301
308   CONTINUE
      CALL FRTCMS('CLRSCRN')
      RETURN
C
C***************  WRITE  FORMAT  STATEMENTS  **********************
C
351    FORMAT(/,7X,'COEFFICIENTS OF NUMERATOR OF SECOND ',
     *        'ORDER CASCADE COMPENSATOR',/,7X,'ENTER ALL ',
     *        'COEFFICIENTS IN DECIMAL FORMAT')
```

```
353    FORMAT(//,7X,' ENTER COEFFICIENT OF S**(',I2,')')                SEC00490
354    FORMAT(/,' COEFFICIENT OF  S**',I2,' :        ',E14.6)            SEC00500
355    FORMAT(//,7X,' ARE THE ABOVE COEFFICIENTS CORRECT? (Y/N)')        SEC00510
356    FORMAT(//,7X,' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY',    SEC00520
     *        ' NUMBER.',/,7X,' TRY AGAIN.')                              SEC00530
C                                                                        SEC00540
C***************  READ  FORMAT  STATEMENTS  ************************** SEC00550
C                                                                        SEC00560
378    FORMAT(F14.6)                                                     SEC00570
379    FORMAT(A1)                                                        SEC00580
C                                                                        SEC00590
       END                                                               SEC00600
```

```
      SUBROUTINE SECASD(CD)
C
C     ****************************************
C     *  SUBROUTINE TO INPUT THE DENOMINATOR   *
C     *  OF THE SECOND ORDER FILTER            *
C     ****************************************
C
C     *** VARIABLE DECLARATIONS ***
      INTEGER ANSWER,YES
      REAL CD
      DIMENSION CD(3)
      DATA YES/'Y'/
C
C
      CALL FRTCMS('CLRSCRN')
401   CONTINUE
      WRITE(6,451)
      DO 406 I=1,3
      J=I-1
404   WRITE (6,453) J
      READ (5,478,END = 405) CD(I)
      GO TO 406
405   REWIND 5
      CALL FRTCMS ('CLRSCRN')
      WRITE(5,456)
      GO TO 404
406   CONTINUE
      CALL FRTCMS('CLRSCRN')
      DO 407 I  = 1,3
       J = 3-I
       K = J+1
       WRITE(6,454) J,CD(K)
407    CONTINUE
       WRITE(6,455)
       READ(5,479)ANSWER
       IF (ANSWER.EQ.YES) GO TO 408
       GO TO 401
408    CONTINUE
       CALL FRTCMS('CLRSCRN')
       RETURN
C
C**************  WRITE  FORMAT  STATEMENTS  **********************
C
451    FORMAT(/,7X,'COEFFICIENTS OF DENOMINATOR OF SECOND ORDER',
     *         ' FILTER',/,7X,'ENTER ALL COEFFICIENTS IN',
     *         ' DECIMAL FORMAT')
453    FORMAT(//,7X,' ENTER COEFFICIENT OF S**(',I2,')')
454    FORMAT(/,' COEFFICIENT OF  S**',I2,' :        ',E14.6)
```

```
455    FCRMAT(//,7X,' ARE THE ABOVE COEFFICIENTS CORRECT? (Y/N)')
456    FORMAT(//,7X,' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY',
     *        ' NUMBER.',/,7X,' TRY AGAIN.')
C
C*************** READ  FORMAT  STATEMENTS  **************************
C
478    FCRMAT(F14.0)
479    FORMAT(A1)
C
       END
```

```
      SUBROUTINE FEEDBK(FK,FN,DN,FP)
C
C     ****************************************
C     *  SUBROUTINE TO INPUT THE PARAMETERS  *
C     *     OF THE FEEDBACK COMPENSATOR      *
C     ****************************************
C
C     *** VARIABLE DECLARATIONS ***
C
      INTEGER FN,DN,N,ANSWER,YES
      REAL FK,FP
      DATA YES/'Y'/
C
      CALL FRTCMS('CLRSCRN')
      WRITE(6,551)
      READ(5,576)N
      CALL FRTCMS('CLRSCRN')
500   CONTINUE
      IF( N .EQ. 3)GO TO 501
      FN=N
      DN=0
      FP=0.0
      WRITE(6,552)
      READ(5,577)FK
      WRITE(6,554)FK
      READ(5,578)ANSWER
      IF(ANSWER .EQ. YES)GO TO 502
      GO TO 500
501   CONTINUE
      FN=2
      DN=1
      WRITE(6,552)
      READ(5,577)FK
      WRITE(6,553)
      READ(5,577)FP
      WRITE(6,555)FK,FP
      READ(5,578)ANSWER
      IF(.NOT.ANSWER.EQ.YES) GO TO 501
502   CONTINUE
      RETURN
C
C**************  WRITE  FORMAT  STATEMENTS  **********************
C
551   FORMAT(//,7X,'YOU MAY USE ONE OF THE FOLLOWING THREE TYPES OF',
     * ' FEEDBACK COMPENSATION',//,7X,
     * '1)   VELOCITY FEEDBACK OF THE FORM :         KT * S',//,7X,
     * '2)   ACCELERATION FEEDBACK OF THE FORM :     KA * (S**2)',//,7X,
     * '3)   APPROXIMATE ACCELERATION FEEDBACK OF THE FORM',/,10X,
```

```
     * '              KA * (S**2)',/,10X,
     * '         ________________',/,10X,
     * '             (S  +  P)',//,7X,
     * ' ENTER 1, 2, OR 3 AS REQUIRED')
552   FORMAT(//,7X,' ENTER VALUE OF CONSTANT K')
553   FORMAT(//,7X,' ENTER VALUE OF POLE P')
554   FORMAT(/,7X,'K =',E14.6,7X,'CORRECT?    (Y/N)')
555   FORMAT(/,7X,'K =',E14.6,7X,'P = ',E14.6,5X,'CORRECT?  (Y/N)')
C
C
C**************  READ  FORMAT  STATEMENTS  *************************
C
576   FORMAT(I1)
577   FORMAT(F14.6)
578   FORMAT(A1)
      END
```

```
      SUBROUTINE FETCH (FENUM)
C
C     ****************************************
C     *  SUBROUTINE TO INPUT THE LOWER LIMIT   *
C     *  OF THE FREQUENCY AXIS.                *
C     ****************************************
C
C     *** VARIABLE DECLARATIONS ***
      INTEGER FENUM,YES,ANSWER
      DATA YES/'Y'/
C
 01   CONTINUE
      WRITE (6,51)
      READ (5,76,END = 02) FENUM
      CALL FRTCMS ('CLRSCRN')
      GO TO 03
 02   REWIND 5
      CALL FRTCMS ('CLRSCRN')
      WRITE(6,52)
      GO TO 01
 03   CONTINUE
      WRITE(6,53)FENUM
      READ(5,77)ANSWER
      IF(ANSWER.EQ.YES)GO TO 04
      GO TO 01
 04   CONTINUE
      RETURN
C
C**************** WRITE  FORMAT  STATEMENTS ***********************
C
 51   FORMAT(' '///,4X,' ENTER THE LOWER LIMIT OF THE FREQUENCY AXIS',
     *        ' AS FOLLOWS :',//,26X,' FOR EXAMPLE   ',//,8X,
     *        'IF LOWER LIMIT IS 10**(-02), ENTER -02',//,8X,
     *        'IF LOWER LIMIT IS 10**(+11), ENTER +11')
 52   FORMAT(///,7X,' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY ',
     *        'NUMBER.',//,7X,' TRY  AGAIN.')
 53   FORMAT(/,4X,'THE LOWER LIMIT IS 10**',I3,'   CORRECT? (Y/N)')
C
C***************** READ  FORMAT  STATEMENTS ************************
C
 76   FORMAT(I3)
 77   FORMAT(A1)
C
      END
```

```
      SUBROUTINE DECADE(DEC)
C
C     ****************************************
C     *  SUBROUTINE TO  INPUT  THE NUMBER OF  *
C     *  DECADES OF FREQUENCY TO BE SPANNED.  *
C     ****************************************
C
C     *** VARIABLE DECLARATIONS ***
      INTEGER DEC,ANSWER,YES
      DATA YES/'Y'/
C
101   CONTINUE
      WRITE (6,151)
      READ (5,176,END=102)DEC
      CALL FRTCMS ('CLRSCRN')
      GO TO 103
102   REWIND 5
      CALL FRTCMS ('CLRSCRN')
      WRITE(6,152)
      GO TO 101
103   CONTINUE
      WRITE(6,153)DEC
      READ(5,177)ANSWER
      IF(ANSWER.EQ.YES) GO TO 104
      GO TO 101
104   CONTINUE
      RETURN
C
C************** WRITE  FORMAT  STATEMENTS ***************************
C
151   FORMAT(' '///,7X,' ENTER AS A SINGLE DIGIT INTEGER, THE NUMBE',
     *         'R OF DECADES OF ',/,7X,' FREQUENCY TO BE SPANNED. ',
     *         /,7X,' MAXIMUM  :  9          MINIMUM  : 1',//)
152   FORMAT(///,7X,' YOU HAVE PRESSED "ENTER" WITHOUT ENTERING ANY ',
     *         'NUMBER.',//,7X,' TRY  AGAIN.')
153   FORMAT(///,7X,I1,' DECADES OF FREQUENCY WILL BE SPANNED',
     *        5X,' CORRECT? (Y/N)')
C
C************* READ  FORMAT  STATEMENTS ******************************
C
176   FORMAT(I1)
177   FORMAT(A1)
C
      END
C
C
```

```
      SUBROUTINE TITLES(TTL1,MSS1)
C
C     ************************************
C     *  SUBROUTINE TO INPUT TWO LINES    *
C     *  AS HEADING FOR THE BODE PLOTS    *
C     ************************************
C
C********  DECLARATION OF VARIABLES  ***************
C
      INTEGER TTL1,MSS1
      DIMENSION TTL1(5),MSS1(5)
      WRITE(6,651)
      READ(5,676)TTL1(1),TTL1(2),TTL1(3),TTL1(4),TTL1(5)
      WRITE(6,653)
      READ(5,676)MSS1(1),MSS1(2),MSS1(3),MSS1(4),MSS1(5)
      RETURN
C
C*********  WRITE  FORMAT  STATEMENTS  ************
C
651   FORMAT(//,7X,'YOU MAY WRITE TWO LINES OF TEXT AS HEADING ON THE',
     *' BODE PLOT',//,7X,'ENTER FIRST LINE (MAXIMUM 20 CHARACTERS)')
653   FORMAT(/,7X,'ENTER SECOND LINE OF TEXT (MAXIMUM 20 CHARACTERS)')
C
C**********  READ  FORMAT  STATEMENT**************
C
676   FORMAT(A4,A4,A4,A4,A4)
C
      END
```

```
C     *********************************************
C     *                                           *
C     *           THE  MAIN  PROGRAM              *
C     *                                           *
C     *********************************************
C
C     ******* DECLARE AND INITIALIZE VARIABLES  *******
C
C
      COMPLEX S,N,D,W,H,NFBK, DFBK,  HFBK,HPLANT,NCAS,DCAS,HCAS
      COMPLEX SECFIL,NCAS2,DCAS2
      REAL HMAG(271),FREQ(271),PHASE(271),LOGW(271),Z,RE,AI,FK,FP
      REAL K,A(10),B(10),CZ(6),CP(6),CN1(3),CN2(3),CD1(3),CD2(3)
      INTEGER NUM,DECS,ANSWER,YES,ANS,CHL,CHC,CHT,CHD,ANS1,TTL(5)
      INTEGER MSS(5),IPAK(28),FN,DN,CHF,CHW,CHN,CHS,CHP,OPEN,ANS2
      DATA YES,Z,CHW,CHL,CHN,CHC,CHT/'Y',0.0,'W','L','N','C','T'/
      DATA CHD,CHS,CHF,CHP,OPEN/'D','S','F','P','O'/

      DATA MSS(1),MSS(2),MSS(3),MSS(4),MSS(5)/' ',' ',' ',' ',' '/
      WRITE(6,1072)
 1025 CONTINUE
      DO 1005 I = 1,6
      CZ(I)=1.0
      CP(I)=1.0
 1005 CONTINUE
      DO 1100 I =1,6
      CN1(I)=1.0
      CN2(I)=1.0
      CD1(I)=1.0
      CD2(I)=1.0
 1100 CONTINUE
      FK=0.0
      FP=0.0
      FN=0
      DN=0
C
C     *******  CALL REQUIRED SUBROUTINES  *******
C
      CALL NUMER (A)
      CALL DENOM (B)
      WRITE(6,1052)
      READ(5,1077)ANSWER
      IF(.NOT.ANSWER.EQ.YES) GO TO 1001
      CALL CASCAD(CZ,CP)
 1001 CONTINUE
      WRITE(6,1056)
      READ(5,1077)ANSWER
      IF(.NOT.ANSWER.EQ.YES)GO TO 1002
```

```
      CALL SECAS(CN1)
      CALL SECASD(CD1)
 1002 CONTINUE
      WRITE(6,1060)
      READ(5,1077)ANSWER
      IF(.NOT.ANSWER.EQ.YES)GO TO 1019
      CALL FEEDBK(FK,FN,DN,FP)
 1019 CONTINUE
      CALL FETCH (NUM)
      CALL DECADE(DECS)
      CALL TITLES(TTL,MSS)
 1014 CONTINUE
      ADECS = FLOAT(DECS)
      WRITE(6,1070)
      READ(5,1076)ANS2
      DO 10 I = 1,271
      LOGW(I) = (I-1)/270.*ADECS
      FREQ(I) = (10.0**(LOGW(I)))*(10.**(NUM))
      S = CMPLX(Z,FREQ(I))
C
C************  UNCOMPENSATED   SYSTEM : NUMERATOR  *****************
C
      N = A(1)*(S**0) + A(2)*(S**1) + A(3)*(S**2) + A(4)*(S**3)
     * + A(5)*(S**4) + A(6)*(S**5) + A(7)*(S**6) + A(8)*(S**7)
     * + A(9)*(S**8) + A(10)*(S**9)
C
C************  UNCOMPENSATED  SYSTEM : DENOMINATOR  *****************
C
      D = B(1)*(S**0) + B(2)*(S**1) + B(3)*(S**2) + B(4)*(S**3)
     * + B(4)*(S**3) + B(5)*(S**4) + B(6)*(S**5) + B(7)*(S**6)
     * + B(8)*(S**7) + B(9)*(S**8) + B(10)*(S**9)
      HPLANT = N/D
C
C     *********  FIRST ORDER CASCADE  FILTERS  **********
C
      NCAS = ((S/CZ(1))+1.)*((S/CZ(2))+1.)*((S/CZ(3))+1.)*
     * ((S/CZ(4))+1.)*((S/CZ(5))+1.)*((S/CZ(6))+1.)
C
      DCAS = ((S/CP(1))+1.)*((S/CP(2))+1.)*((S/CP(3))+1.)*
     * ((S/CP(4))+1.)*((S/CP(5))+1.)*((S/CP(6))+1.)
C
      HCAS = NCAS/DCAS
C
C************  SECOND ORDER CASCADE FILTERS *************
      NCAS2 =(CN1(3)*(S**2)+CN1(2)*(S**1)+CN1(1))
C    *       *(CN2(3)*(S**2)+CN2(2)*(S**1)+CN2(1))
      DCAS2 =(CD1(3)*(S**2)+CD1(2)*(S**1)+CD1(1))
C    *       *(CD2(3)*(S**2)+CD2(2)*(S**1)+CD2(1))
```

```
      SECFIL=NCAS2/DCAS2
C
C     *************** FEEDBACK COMPENSATION ************
C
      NFBK = FK*(S**FN)
      DFBK = (S**DN)+(FP)
      HFBK = NFBK/DFBK
      H = HPLANT*HCAS*SECFIL
      H = H/(1.0 +(HFBK*H))
      IF(ANS2 .EQ. OPEN) GO TO 1128
      H = H/(1.0 + H)
1128  CONTINUE
      RE = REAL(H)
      AI = AIMAG(H)
      PHD =57.295 *  ATAN2(AI,RE)
      IF (PHD .GT.0.) PHD = PHD - 360.
      PHASE(I) = PHD
      HMAG(I) = 20.*ALOG10(CABS(H))
C     IF(HMAG(I).GT.120.)HMAG(I) = 120.
C     IF(HMAG(I) .LT.-120.)HMAG(I)= -120.
10    CONTINUE
      WRITE(6,1067)
      READ(5,1076)ANSWER
      IF(.NOT. ANSWER .EQ. YES) GO TO 1999
      CALL FRTCMS('CLRSCRN')
      WRITE(6,1065)
      DO 1999 I=1,271,3
      WRITE(6,1066)FREQ(I),HMAG(I),PHASE(I)
1999  CONTINUE
      CALL TEK618
C     CALL BLOWUP(1.2)
      CALL PAGE(11.,8.5)
      CALL NOCHEK
      CALL AREA2D(9.,6.)
      CALL XNAME('FREQUENCY (RAD. PER SEC)$',100)
      CALL YNAME('TRANSFER FUNCTION (MAG)$',100)
      CALL HEADIN(TTL,20,1.5,2)
      CALL HEADIN(MSS,20,1.5,2)
      CALL XLOG(FREQ(1),9./ADECS,-60.00,20.)
      CALL LINES('PHASE$',IPAK,2)
      CALL LINES('GAIN$',IPAK,1)
      XW=XLEGND(IPAK,2)
      YW=YLEGND(IPAK,2)
      XL=8.-.5-XW-.1
      YL=6.-.5-YW-.1
      CALL BLNK1(XL-.3,8.-.4,YL-.3,6.-.4,2)
      CALL LINESP(3.5)
      CALL DOT
```

```
      CALL GRID(1,2)                                                 MAI01450
      CALL RESET('DOT')                                              MAI01460
      CALL THKCRV(.015)                                              MAI01470
      CALL LEGLIN                                                    MAI01480
      CALL CURVE(FREQ,HMAG,271,0)                                    MAI01490
      CALL YGRAXS(-360.,60.,0.,5.,'PHASE$',-100,9.0,0.)              MAI01500
      CALL DOT                                                       MAI01510
      CALL LEGLIN                                                    MAI01520
      CALL CURVE(FREQ,PHASE,271,0)                                   MAI01530
      CALL RESET('BLNK1')                                            MAI01540
      CALL RESET('THKCRV')                                           MAI01550
      CALL LEGEND(IPAK,2,XL,YL)                                      MAI01560
      CALL ENDPL(0)                                                  MAI01570
      WRITE(6,1051)                                                  MAI01580
      READ(5,1076) ANSWER                                            MAI01590
      IF(.NOT.ANSWER .EQ. YES) GO TO 1006                            MAI01600
                                                                     MAI01610
1018  CONTINUE                                                       MAI01620
      WRITE(6,1053)                                                  MAI01630
      READ (5,1078)ANS                                               MAI01640
      CALL FRTCMS('CLRSCRN')                                         MAI01650
      IF(ANS.EQ.CHP)GO TO 1014                                       MAI01660
      IF(ANS.EQ.CHW)GO TO 1010                                       MAI01670
      IF(ANS.EQ.CHL)GO TO 1011                                       MAI01680
      IF(ANS.EQ.CHC)GO TO 1012                                       MAI01690
      IF(ANS.EQ.CHT)GO TO 1017                                       MAI01700
      IF(ANS.EQ.CHN)GO TO 1124                                       MAI01710
      IF(ANS.EQ.CHD)GO TO 1125                                       MAI01720
      IF(ANS.EQ.CHS)GO TO 1126                                       MAI01730
      IF(ANS.EQ.CHF)GO TO 1127                                       MAI01740
1010  CALL DECADE(DECS)                                              MAI01750
      GO TO 1015                                                     MAI01760
1011  CALL FETCH(NUM)                                                MAI01770
      GO TO 1015                                                     MAI01780
1012  CALL CASCAD(CZ,CP)                                             MAI01790
      GO TO 1015                                                     MAI01800
1017  CALL TITLES(TTL,MSS)                                           MAI01810
      GO TO 1015                                                     MAI01820
1124  CALL NUMER(A)                                                  MAI01830
      GO TO 1015                                                     MAI01840
1125  CALL DENUM(B)                                                  MAI01850
      GO TO 1015                                                     MAI01860
1126  CALL SECAS(CN1)                                                MAI01870
      CALL SECASD(CD1)                                               MAI01880
      GO TO 1015                                                     MAI01890
1127  CALL FEEDBK(FK,FN,D,FP)                                        MAI01900
1015  WRITE(6,1055)                                                  MAI01910
      READ(5,1078)ANS1                                               MAI01920
```

```
      IF(.NOT.ANS1.EQ.YES)GO TO 1014
      GO TO 1018
 1006 CONTINUE
      WRITE(6,1057)
      READ(5,1076)ANSWER
      IF(ANSWER.EQ.YES)GO TO 1025
      CALL DONEPL
      STOP
 1051 FORMAT(///,7X,' DO YOU WISH TO HAVE ANOTHER RUN OF THE SAME',
     * ' PROBLEM   (Y/N)')
 1053 FORMAT(/,7X,'ENTER CATERGORIES FOR CHANGES AS FOLLOWS :',/,7X,
     * 'N    CHANGE IN NUMERATOR OF PLANT TRANSFER FUNCTION',/,7X,
     * 'D    CHANGE IN DENOMINATOR OF PLANT TRANSFER FUNCTION',/,7X,
     * 'W    NUMBER OF DECADES TO BE SPANNED',/,7X,
     * 'L    LOWER LIMIT OF FREQUENCY AXIS',/,7X,
     * 'C    CHANGE IN FIRST ORDER CASCADE FILTERS',/,7X,
     * 'T    TITLES',/,7X,
     * 'S    CHANGE IN SECOND ORDER CASCADE FILTER',/,7X,
     * 'F    CHANGE IN FEEDBACK COMPENSATION',/,7X,
     * 'P    NO CHANGES')
 1052 FORMAT(/,7X,'ANY FIRST ORDER CASCADE COMPENSATORS? (Y/N)')
 1055 FORMAT(/,7X,'ANY MORE CHANGES?  (Y/N)')
 1056 FORMAT(/,7X,'ANY SECOND ORDER CADCADE FILTERS?    (Y/N)')
 1057 FORMAT(//,7X,'DO YOU WISH TO RUN A NEW PROBLEM?  (Y/N)')
 1060 FORMAT(/,7X,'ANY FEEDBACK COMPENSATION?  (Y/N)')
 1065 FORMAT(/,7X,'FREQ',15X,'MAGNITUDE',10X,'PHASE',///,)
 1066 FORMAT(4X,E14.6,5X,E14.6,5X,E14.6)
 1067 FORMAT(/,7X,'DO YOU WANT TABULAR OUTPUT AT THE TERMINAL?',
     *        4X,'(Y/N)')
 1070 FORMAT(/,7X,'DO YOU WANT OPEN OR CLOSE LOOP RESPONSE?',//,7X,
     *        'ENTER  O  FOR OPEN LOOP',/,12X,'  C  FOR CLOSE LOOP')
 1072 FORMAT(///,7X,'BODPLT IS AN EASY, INTERACTIVE METHOD TO PRODUCE',
     *        ' OPEN AND CLOSE LOOP',/,7X,'BODE PLOTS.  THE TRANSFER',
     *        ' FUNCTION DATA AND PARAMETERS OF CASCADE AND/OR',/,7X,
     *        'FEEDBACK COMPENSATORS ARE TO BE ENTERED MANUALLY',
     *        ' AT THE KEYBOARD.',/,7X,'IF YOU WANT A PRINT-OUT OF',
     *        ' THE TABULAR RESULTS, USE THE "RECORD ON"',/,7X,
     *        '"RECORD OFF" EXECS. YOU SHOULD BE LOGGED ON AT THE',
     *        ' IBM 3277/TEK 618',/,7X,'DUAL SCREEN TERMINALS.',
     *        ////////,'.')
 1076 FORMAT(A1)
 1077 FORMAT(A1)
 1078 FORMAT(A1)
C
      END
```

LIST OF REFERENCES

1. Gianniotis, Demetrios D., Interactive Control System Optimization and Simulation, Thesis Dissertation, Naval Postgraduate School, October 1982.

2. Thaler, G.J., Design of Feedback Systems, Dowden Hutchinson and Ross, Inc., 1973

3. Ward, J.R., and Strum, R.D., State Variable Analysis, Naval Postgraduate School, Monterey, California, 1975.

4. Luenberger, D.G., Introduction to Linear and Nonlinear Programming, Addison-Wesley Publishing Company, Inc. 1973.

5. Kuo, Benjamin C., Automatic Control Systems, Prentice-Hall, Inc., 1982.

6. Thaler, G.J., Automatic Control Systems, Naval Postgraduate School, 1984.

7. Bazarra, M.S., Nonlinear Programming, Theory and Algorithms, John Wiley and Sons, Inc., 1979

8. Box, M.J., "A New Method of Constraint Optimization and a Comparison with Other Methods," Computer Journal, pp. 45-52, 8 April 1965.

9. Schaum's Outline Series, Programming With Fortran, McGraw-Hill Book Company, 1978.

10. W.R. Church Computer Center, Technical Note, "User's Guide to VM/CMS at NPS," April 1981.

11. W.R. Church Computer Center, Technical Note, "Using Disspla at NPS," June 1984.

12. Integrated Software Systems Corporation, "Disspla Users' Manual," October 1981.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
2.	Professor George J. Thaler, Code 62TR Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943	5
3.	Professor Alex Gerba, Jr., Code 62GZ Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943	1
4.	Department Chairman, Code 62RR Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943	1
5.	Naval Secretary Naval Headquarters Islamabad Pakistan	1
6.	Directorate of Naval Training Naval Headquarters Islamabad Pakistan	1
7.	Directorate of Naval Weapons and Equipment Naval Headquarters Islamabad Pakistan.	1
8.	Commander Karachi 9 Liaquat Barracks Karachi Pakistan	1

9. Commanding Officer — 1
Pakistan Naval Engineering College
Habib Ibrahim Rahimtoola Road
Karachi
Pakistan

10. LCDR Habib Ismail, P.N. — 5
Pakistan Naval Engineering College
PNS Jauhar
Habib Ibrahim Rahimtoola Road
Karachi
Pakistan

11. LCDR M. Tariq, P.N. — 1
Pakistan Naval Engineering College
PNS Jauhar
Habib Ibrahim Rahimtoola Road
Karachi
Pakistan

12. Tayfun Tansan — 1
Goztepe Istasyon Cad. Np. 31/13
Istanbul
Turkey

13. LT Kyritsis-Spromilios Pericles, H.N. — 1
Sarantaporou 125
Papagou
Athens
Greece

14. LT Horianopoulos Emmanuel, H.N. — 1
Travlantoni 39
Patisia
Athens
Greece

15. Defense Technical Information Center — 2
Cameron Station
Alexandria, Virginia 22314

Thesis
I727 Ismail
c.1 Interactive computer
program for the analysis and design of
linear time invariant
systems.